Discussion of Midterm Questions

1. Exercise 4: An Exercise on the Definite Article *The*

Background:
It looks like the meaning of *the* requires that its NP complement be true of exactly one entity *in some contextually determined set C of 'contextually salient entities'.*

How do we write a semantic entry for *the* that captures this idea?

(1) **The Targeted Solution**

[[the]]$^{g, C}$ = [λf : f $\in$ $D_{<et>}$ and there is exactly one x **in C** such that f(x) = T .
the unique y in C such that f(y) = T]

You were then asked to compute the extension of the DP *the dog* in two different contexts. The main lesson of doing so is the following.

(2) **The Upshot**

Given the semantics in (1), the DP *the dog*:

- Only has a value if the context makes salient exactly one dog
  (if there is exactly one dog in C)

- Will vary in its extension depending upon the exact identity of the one dog in C.
  If C = { Dave, Scooby-Doo }, then [[the dog]]^{C} = Scooby Doo
  If C = { Dave, Scrappy-Doo], then [[the dog]]^{C} = Scrappy Doo

(3) **Question**
Is there any broader use of C beyond this issue relating to definite descriptions?...

(4) **Observations**

Suppose we are at a party…

a. "Every woman is dancing" is T *iff* all the woman *at the party* are dancing.

b. "Some woman is dancing" is T *iff* some woman *at the party* is dancing.

c. "No woman is dancing" is T *iff* no woman *at the party* is dancing.

(5) **General Observation**

It seems that quantification in natural language is also restricted by this parameter 'C'

(6) **Revised Lexical Entries**

a. [[every]]^C = [$\lambda f_{<et>}$: $\lambda g_{<et>}$: for all **x in C**, if f(x) = T, then g(x) = T]]

b. [[some]]^C = [$\lambda f_{<et>}$: $\lambda g_{<et>}$: there is some **x in C** such that f(x) = T and g(x) = T]]

c. [[no]]^C = [$\lambda f_{<et>}$: $\lambda g_{<et>}$: there is no **x in C** such that f(x) = T and g(x) = T]]

(7) **Question**

Are we missing something by stipulating the sensitivity to C in *each* of these different quantifiers…
…what if it's really a more basic feature of predicates, things of type <et>

[[dog]]^C = [λx : **x in C** and x is a dog]

---

2. Exercise 6: Binding and Gender

(8) **The Over-Arching Question**
Is our treatment of bound pronouns consistent with our treatment of pronominal gender?

(9) **Question 3**
Can sentence (a) be shown to have the T-conditions in (b)?

a. Dave is a man who_1 loves $himself_1$.
b. Dave is a man and Dave loves Dave.

(10) **Answer: YES!**

a. [[Dave is a [man [who_1 [t_1 loves $himself_1$]]]]]^g = T *iff* (by FA, PM, NN, TN)

b. [λx : x is a man and [λz : [[t_1 loves $himself_1$]]$^{g(1/z)}$ = T](x) = T](Dave) = T *iff*

c. [λx : x is a man and [[t_1 loves $himself_1$]]$^{g(1/x)}$ = T](Dave) = T *iff*

d. Dave is a man and [[t_1 loves $himself_1$]]$^{g(1/Dave)}$ = T *iff* (by FA, PR, notation)

e. Dave is a man and Dave loves [[MASC pro_1]]$^{g(1/Dave)}$ *iff* (by FA, TN, PR)

f. Dave is a man and Dave loves [λx : x is a man . x](Dave) *iff* (by LC)

g. Dave is a man and Dave loves Dave

(11) **Observation**

- The computation in (10) is completely unproblematic.
  - Every step in the computation is licensed by our rule system.
  - At no step do we have to ask any 'confusing questions'
- If we follow the computation in (10), we also get an answer to the following question.

(12) **Question 4**
Can sentence (a) be assigned the T-conditions in (b)?

a. Dave is a man who_1 loves herself_1.
b. Dave is a man and Dave loves Dave.

(13) **Answer: NO!**

a. $[[\text{ Dave is a [man [who}_1 \text{ [} t_1 \text{ loves herself}_1 \text{]]]]]}^g = T$ *iff* (by FA, PM, NN, TN)

b. $[\ \lambda x : x \text{ is a man and } [\ \lambda z : [[t_1 \text{ loves herself}_1]]^{g(1/z)} = T](x) = T\ ](\text{Dave}) = T$ *iff*

c. $[\ \lambda x : x \text{ is a man and } [[t_1 \text{ loves herself}_1]]^{g(1/x)} = T\ ](\text{Dave}) = T$ *iff*

d. Dave is a man and $[[\ t_1 \text{ loves herself}_1]]^{g(1/\text{Dave})} = T$ *iff* (by FA, PR, notation)

e. Dave is a man and Dave loves $[[\text{ FEM } pro_1\ ]]^{g(1/\text{Dave})}$ *iff* …

STOP!

- $[[\ pro_1\ ]]^{g(1/\text{Dave})}$ is *not* in the domain of [[FEM]]
- So, no semantic value can be derived for $[[\text{ FEM } pro_1\ ]]^{g(1/\text{Dave})}$
- So, *the sentence in (12a) is uninterpretable, and so ill-formed!*

So, if we follow the computation in (10) and (13), we obtain sensible, empirically correct answers to the questions in (9) and (12)…
… but *nobody* followed this computation…
… *everybody* did something slightly different, which (should have) forced them to ask a very difficult and confusing question…

(14) **Confession Time**

I hoped you would follow this other computation, so that you *would* be forced to ask this confusing question, so that I make a subtle-though-crucial point about variable assignments like '$g(1/x)$' and what they *are*…

(15) **Alternative Computation**

a. [[Dave is a [man [who$_1$ [t_1 loves himself$_1$]]]]]^g = T *iff* (by FA, PM, NN, TN)

b. [λx : x is a man and [λz : [[t_1 loves himself$_1$]]$^{g(1/z)}$ = T](x) = T](Dave) = T *iff*

c. [λx : x is a man and [[t_1 loves himself$_1$]]$^{g(1/x)}$ = T](Dave) = T *iff* (by tons of rules)

d. [λx : x is a man and x loves [[MASC *pro*$_1$]]$^{g(1/x)}$](Dave) = T *iff* …

… and at this point, everyone felt they were forced to ask a very confusing question…

(16) **The Confusing Question**

- Is [[*pro*$_1$]]$^{g(1/x)}$ in the domain of [[MASC]]?
- That is, since [[*pro*$_1$]]$^{g(1/x)}$ = x (by PR), then is x in the domain of [[MASC]]?

This question in (16) is confusing, because in order for the question to make any sense at all, you have to *very careful about how you ask it …*

… and in answering the question, it's very important that you understand what the question ***means…***

(17) **The Wrong Way to Ask This Question**

- Is *the variable* 'x' in the domain of [[MASC]]?
- Does *the variable* 'x' count as male somehow?

(18) **Negative Consequences of Asking the Question This Way**

a. *Suppose that we don't allow 'the variable x' to be in the domain of* [[MASC]].

PROBLEM:
We can't compute an interpretation for *Dave is a man who loves himself.*

a. *Suppose we allow variables to be in the domain of [[MASC]], because 'variables are underspecified for gender'.*

PROBLEM:
We can now compute an interpretation for *Dave is a man who loves herself*, since:
[λx : x is a man and x loves [[FEM *pro*$_1$]]$^{g(1/x)}$] =
[λx : x is a man and x loves x] =

(19) **What's Ultimately Wrong with the Questions in (17)?**

The questions in (17) assume that 'x' in our metalanguage *refers* to **the variable x**.

But this isn't so:
- 'x' in our metalanguage *is* the variable 'x'
- 'x' in our metalanguage *refers* to some entity

<u>A Useful Analogy:</u>
- 'Obama' in our metalanguage *is* the name 'Obama'
- 'Obama' in our metalanguage refers to Obama
  (it doesn't refer to the *name* Obama, which is itself)

If the point above is confusing, that's because it is…
… but it starts to make more sense if you look beyond the ***form*** *of our lambda expressions and think about what they mean…*

(20) **A Very Useful Set of Paraphrases**

a. <u>The Lambda Notation</u>

[λx : x is a man and x loves [[MASC pro_1]]$^{g(1/x)}$]

b. <u>What the Notation Means, 1</u>

The function which ***for any entity x****, yields T iff*
x *is a man and* ***x*** *loves* [[MASC pro_1]]$^{g(1/\mathbf{x})}$

But, the variables 'x' in (20b) above are just a fancy, unambiguous way of saying "*that thing*", like as follows:

c. <u>What the Notation Means, 2</u>

The function which ***for any entity****, yields T iff*
that thing *is a man and* ***that thing*** *loves*
the interpretation of "MASC pro_1" relative to an assignment
where 1 is mapped to **that thing**

Now, given all this, ask yourself the question in (21), which is really equivalent to the (correct, sensible) question in (16):

(21) **Key Question (= Question (16))**

In the statement in (20c), can I determine whether **that thing** is in the domain of [[MASC]]?

(22) **ANSWER: NO!**

- Whether **that thing** is in the domain of [[MASC]] depends upon whether **that thing** is male or female…
  … and nothing in the statement in (20c) determines this!
  (… all we know about **that thing** from (20c) is that it is an entity…)

(23) **Conclusion**

We can't complete the derivation in (15) by calculating the value of [[MASC pro_1]]$^{g(1/x)}$
However, we can complete the calculation as follows:

(24) **Completion of the Calculation in (15)**

a. [[Dave is a [man [who$_1$ [t_1 loves himself$_1$]]]]]^g = T *iff* (by FA, PM, NN, TN)
b. [λx : x is a man and [λz : [[t_1 loves himself$_1$]]$^{g(1/z)}$ = T](x) = T](Dave) = T *iff*
c. [λx : x is a man and [[t_1 loves himself$_1$]]$^{g(1/x)}$ = T](Dave) = T *iff* (by tons of rules)
d. [λx : x is a man and x loves [[MASC pro_1]]$^{g(1/x)}$](Dave) = T *iff* **(by notation)**
e. Dave is a man and Dave loves [[MASC *pro*$_1$]]$^{g(1/Dave)}$ *iff*
f. Dave is a man and Dave loves Dave

... and when we complete the calculation in this manner, we again predict that "Dave is a woman who loves herself" is uninterpretable...

(25) **Another Key Example** The man who loves himself.

(26) **Question:** Can our system interpret the DP in (25)?

(27) **ANSWER: Yes!**

a. In interpreting (25), our system eventually comes to the following value:

'the unique **y** such that **y** is a man and **y** loves [[MASC pro_1]]$^{g(1/y)}$'

*the unique thing such that **that thing** is a man and **that thing** loves the interpretation of* "MASC pro_1" *relative to an assignment where 1 is mapped to **that thing**.*

b. But here, we know that **y** (**that thing**) has to be a man! So, we know that [[pro_1]]$^{g(1/y)}$ has to be a man, and so we can compute further to get:

the unique **y** such that **y** is a man and **y** loves **y**.

(28) **Consequence** "the man who loves herself" is uninterpretable.

(29) **The Main Issue**

- If we assume that gender features are 'partial identity functions', then it can become tricky interpreting bound pronouns…

- In calculating the value of $[[\text{MASC/FEM pro}_1]]^{g(1/x)}$, you should be mindful of the fact that variables in our metalanguage refer to *things*…(and not 'variables')

- So, to calculate the value of $[[\text{MASC/FEM pro}_1]]^{g(1/x)}$, you need to ask yourself what you know about *the thing* that the variable 'x' could be referring to (in the expression in question)…

(30) **General Consequence**

$[[\text{MASC/FEM pro}_1]]^{g(1/x)}$ can only be computed to have a value (as x) if it's somehow guaranteed/assumed that x is something a male / female…

The consequence in (30) has interesting, observable effects upon the meaning of sentences containing bound pronouns…

(31) **Example 1**

$[[\text{the pilot that loves himself}]]^g$ =

the unique **y** such that **y** is a pilot and **y** loves $[[\text{MASC } pro_1]]^{g(1/\mathbf{y})}$ =

the unique thing such that ***that thing*** *is a pilot and* ***that thing*** *loves* $[[\text{MASC } pro_1]]^{g(1/\textbf{that thing})}$

(32) **Result 1**

- The only way "the pilot that loves himself" can have a value is if **y** (**that thing**) is assumed to be male.

- With that assumption, this DP refers to *the unique y such that y is a pilot and y loves y*

- Thus, this DP necessarily presupposes *that the unique y such that y is a pilot and y loves y* is male.

- *…which conforms to native speaker intuition about that DP!*

(33) **Example 2**

[[every pilot loves himself]]^{g} = T iff

For every **x** such that **x** is a pilot, **x** loves [[MASC *pro*$_1$]]$^{g(1/\mathbf{x})}$ *iff*

For any arbitrary thing such that **that thing** is a pilot, **that thing** loves [[MASC *pro*$_1$]]$^{g(1/\textbf{that thing})}$

(34) **Result 2**

- The only way that "every pilot loves himself" can have a value is if **x** (**that thing**) is assumed to be male.
- *But **x (that thing)** is any arbitrary pilot…and so our assumption must be that any arbitrary pilot is male…*
- With that assumption, the sentence's T-conditions are
  For any x such that x is a pilot, x loves x
- Thus, this sentence necessarily presupposes that *all pilots are male…*
- *…which conforms to native speaker intuition about that DP!...*

(35) **General, Practical Point**

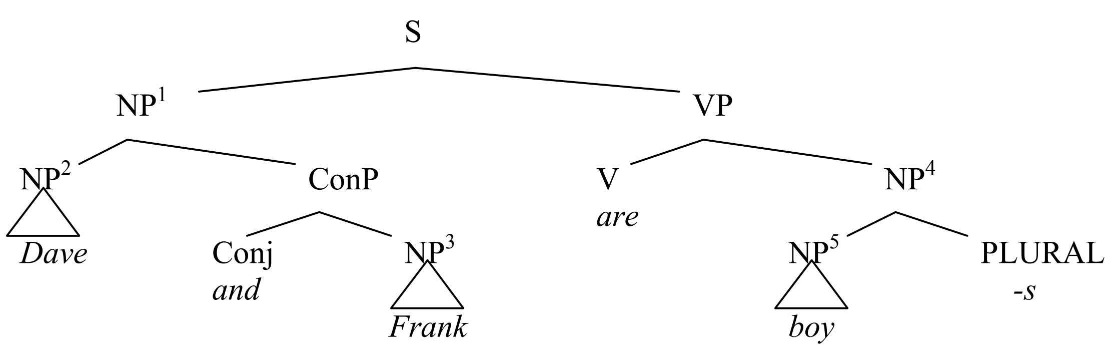
From now on, we can ignore gender in pronouns. *But it's important to note these empirical consequences of our theory of gender as a 'presupposition-trigger'…*

---

3. Exercise 7: An Initial Foray into Plurals

You were asked to assume that the structure of "Dave and Frank are boys" is as follows:

(36)



You were also asked to assume that the coupla in this sentence has exactly the same meaning as the copula in a sentence like "Dave is a boy".

(37) **Question 1**

a. Semantic type of the V: <et,et>

b. Semantic type of the VP: <et>

c. Semantic type of the NP^4: <et>

You were then asked to assume that "boy" in (36) has exactly the same meaning as the NP "boy" in a sentence like "Dave is a boy."

(38) **Question 2**

a. Semantic type of the NP: <et>

b. Semantic type of PLURAL: <et,et>

(39) **Question 3**

Semantic type of NP^1: e

You were then asked to provide a semantics for "and" in (36), which built on the idea that D_e could contain special 'plural' entities called groups (where Frank+Dave is a shorthand for the group formed from Frank and Dave)

(40) **Question 4**

[[and]] = [λx : [λy : x+y]]

You were then asked to give a semantics for PLURAL that would derive the following T-conditions for sentence (36).

(41) **Targeted T-Conditions**

(36) is T *iff* Frank+Dave is a group formed from the members of { x : x is a boy }

(42) **Semantics for Plural**

[[PLURAL]] = [$\lambda f_{<et>}$: [λy : y is a group formed from the members of { x : f(x) = T }]

(43) **Interim Comments**

- This semantics for PLURAL is more-or-less the standard picture regarding the meaning of PLURAL

- This general treatment of PLURAL in natural language goes back (at least) to the work of the philosopher W.V.O. Quine. However, linguists know it best from its formulation by the linguist/logician Godehard Link (Link 1983).

- This semantics predicts a number of 'neat' entailments such as:
  "Dave and Frank are boys" is T *iff* "Dave is a boy" is T and "Frank is a boy" is T

You were then asked to combine this semantics for plural NPs with our semantics for "the".

(44) **Problematic Prediction**

- Under our current semantics, "the" can only take an NP complement if that NP is true of exactly one thing.

- But, if there are at least three boys, then the plural NP boys is true of more than one thing (because you can form more than one group of boys).

- But, it seems like "the boys" does have a meaning in such cases…

(45) **Observation Towards a Solution**

- Suppose that [[boy]] is true of exactly the following: Seth, Rajesh, and Kyle

- In such a context, it seems that "The boys smoke" is T *iff* Seth+Rajesh+Kyle smoke.

  …if only Seth+Rajesh smoked, then the sentence would be false…

(46) **Key Idea**

Maybe "the boys" refers to the largest possible group of boys.

We should refine this by clarifying what we mean by 'largest possible'…

(47) **The 'Part of' Relation Amongst Groups**

- Intuitively, different groups (and individuals) can be part of other groups

- Seth is a part of Seth+Rajesh, and Seth+Rajesh is a part of Seth+Rajesh+Kyle

- We might also assume that for all entities x, x is a part of x itself (Seth is part of Seth)

(47) **Largest Entity**

A set of groups S has a 'largest' entity if there is some entity x in S such that *for all y* in S, y is a part of x.

Such an entity x is called 'the largest entity' in S.

Example 1

- S = {Seth, Rajesh, Kyle, Seth+Rajesh, Seth+Kyle, Kyle+Rajesh, Seth+Kyle+Rajesh}
- S has a largest member: Seth+Kyle+Rajesh

Example 2

- A = {Seth, Rajesh, Kyle, Seth+Rajesh}
- A *doesn't* have a largest member.

With this as background, consider the following semantics for "the"…

(48) **New Semantics for *The***

[[the]] = [λf : f $\in D_{<et>}$ and the set { x : f(x) = T } has a largest entity .
the largest member of { x : f(x) = T }]

This semantics captures the notion that "the boys" should refer to the largest possible set of boys!

(49) **Positive Result**

- Suppose that singular NP [[boy]] is true of exactly Seth, Rajesh and Kyle

- Then the plural NP [[boys]] will be true of exactly:
  Seth, Rajesh, Kyle, Seth+Rajesh, Seth+Kyle, Kyle+Rajesh, Seth+Rajesh+Kyle

- Then the set { x : [[boys]](x) = T } =
  {Seth, Rajesh, Kyle, Seth+Rajesh, Seth+Kyle, Kyle+Rajesh, Seth+Rajesh+Kyle}

- Then [[the]]([[boys]]) = Seth+Rajesh+Kyle (by (47) and (48))

(50) **Crucial Question**

This semantics for "the" works well when its NP complement is *plural*…
…but what about all our earlier cases where its NP complement is *singular*…

(51) **Key Observations**

a. If set consists of just one (atomic) entity, then that set *has a largest entity*.

- Consider the set { Seth }
- Since Seth is a part of Seth, there is some entity x in this set (Seth) such that for all entities y in the set, y is a part of x.

b. If a set consists of at least two (atomic, non-group) entities, then that set *doesn't have a largest entity.*

- Consider the set {Kyle, Seth}
- Since Seth isn't a part of Kyle, and Kyle isn't a part of Set, there is no entity x such that for *all* members y of this set, y is a part of x.

(52) **Key Conclusion**

If an NP is *singular* (and so is T only of atomic entities), then { x : [[NP]](x) = T } has a largest entity *if and only if* there is exactly one x such that [[NP]](x) is T.

(53) **Key Consequence**

Our new semantics in (48) captures all the data that our earlier semantics for "the" was designed to capture…

- If NP is singular, then [[the NP]] has a value *iff* NP is T of *exactly* one entity.
- When [[the NP]] has a value, it refers to the one entity that [[NP]] is true of.

---

3. Some More Fun With Plurals

Nearly all quantificational Ds allow (or require) for their NP complements to be plural.

(54) **Plural Quantificational DPs**

a. Some dogs
b. No dogs
c. Three dogs
d. Many dogs
e. Most dogs

The treatment of quantificational Ds that we've developed in class – and which is developed in most introductory texts – *ignores* the plurality of the NP complements…

… it treats the plurality as if it were not just ignored by the semantic system.

(55) **Illustration: The Semantics of "Two"**

[[two]] = $[\lambda g_{<et>} : [\lambda f_{<et>} : |\{ x : g(x) = T \text{ and } f(x) = T \}| \geq 2]]$

(56) **Observation 1**

The semantics in (55) only predicts the correct T-conditions if we assume that [[two]] takes the extension of *singular* [[dog]] as argument.

a. [[two]]([[dog]])([[smoke]]) = T *iff* | { x : x is **a dog** and x smokes } | ≥ 2

Thus, this semantics must assume that [[dogs]] = [[dog]].

(57) **Observation 2**

If we assume that "two" has the meaning in (55), then if we interpret the plurality on its NP complement, we get absolutely the wrong T-conditions!

b. [[two]]([[dogs]])([[smoke]]) = T *iff*
| { x : x is **a group of dogs** and x smokes } | ≥ 2
There are two groups of dogs that smoke.

(58) **Conclusion**

If we want to assume that plurality in (54) is interpreted (and we should…what else could the plurality here be doing?...)
… then we need a different semantics for determiners like "three"…

(58) **Key Intuition**

- "Two dogs smoke" *isn't* true if two *groups* of dogs smoke…
- "Two dogs smoke" is true *iff* **a group consisting of two dogs smokes…**

(59) **New Semantics for "Two"**

[[two]] = [$\lambda g_{<et>}$: [$\lambda f_{<et>}$: there is **some x** such that g(x) = T and
| { y : y is a part of x } | = 2 and f(x) = T

(60) **Predicted T-Conditions**

[[two]]([[dogs]])([[smoke]]) = T *iff* there is some x such that **x is a group of dogs** and
| { y : y is a part of x } | = 2 and **x smokes**

(61) **General Issue**

- Many of our quantificational Ds in English can actually be thought of as special kinds of 'plural indefinites'…
- These quantificational Ds assert that *there exists* some group *of a particular size…*
- …and the NP and VP then add further information about the nature of this group…